Lecture 07
Deep Networks II

仿生设计
与学习
实验室

Bionic Design
& Learning Lab

AncoraSIR.com

[Please refer to the course website for copyright credits]

SUSTech
Southern University
of Science and Technology

Convolutional Networks

仿生设计与学习实验室 Bionic Design & Learning Lab

AncoraSIR.com

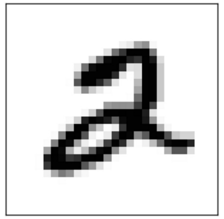
SUSTech Southern University of Science and Technology

A Design Challenge with Increasing Dimensions

Regular Neural Nets don't scale well to full images

$\mathbf{512\times512\times3=765,432}$

Gray-scaled



MNIST

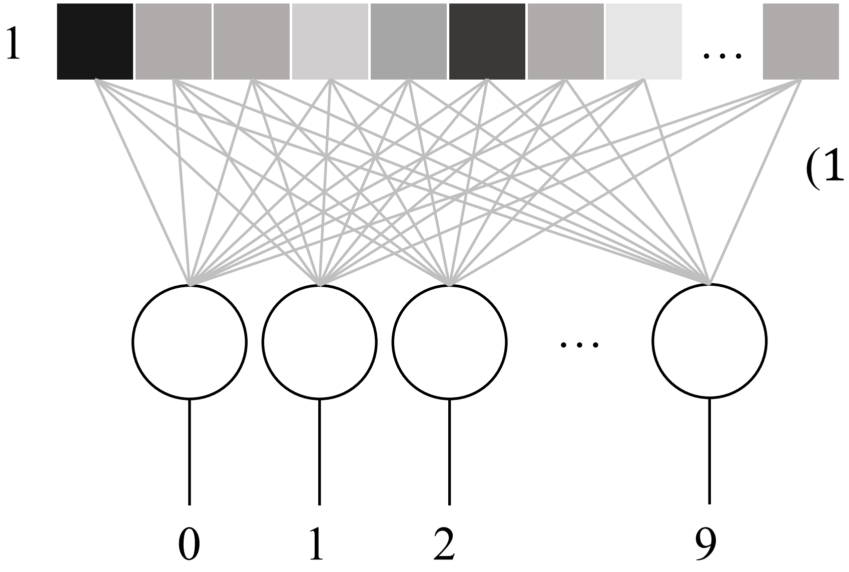
$28\times28\times1=784$

Input *Features*

CIFAR-10

Color (RGB)

$32\times32\times3=3{,}072$



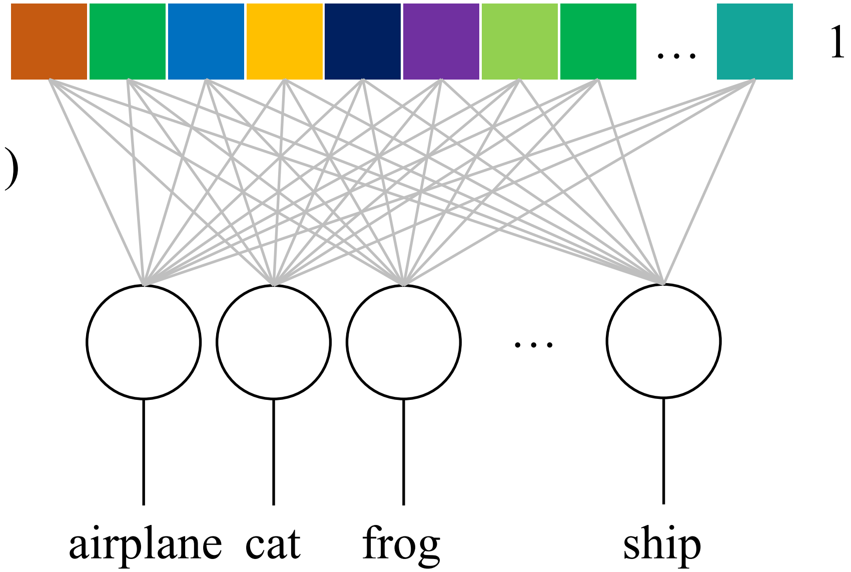
Weighted-Sum

$(10\times784 \leftarrow W \rightarrow 10\times3072)$

Activation

(?)

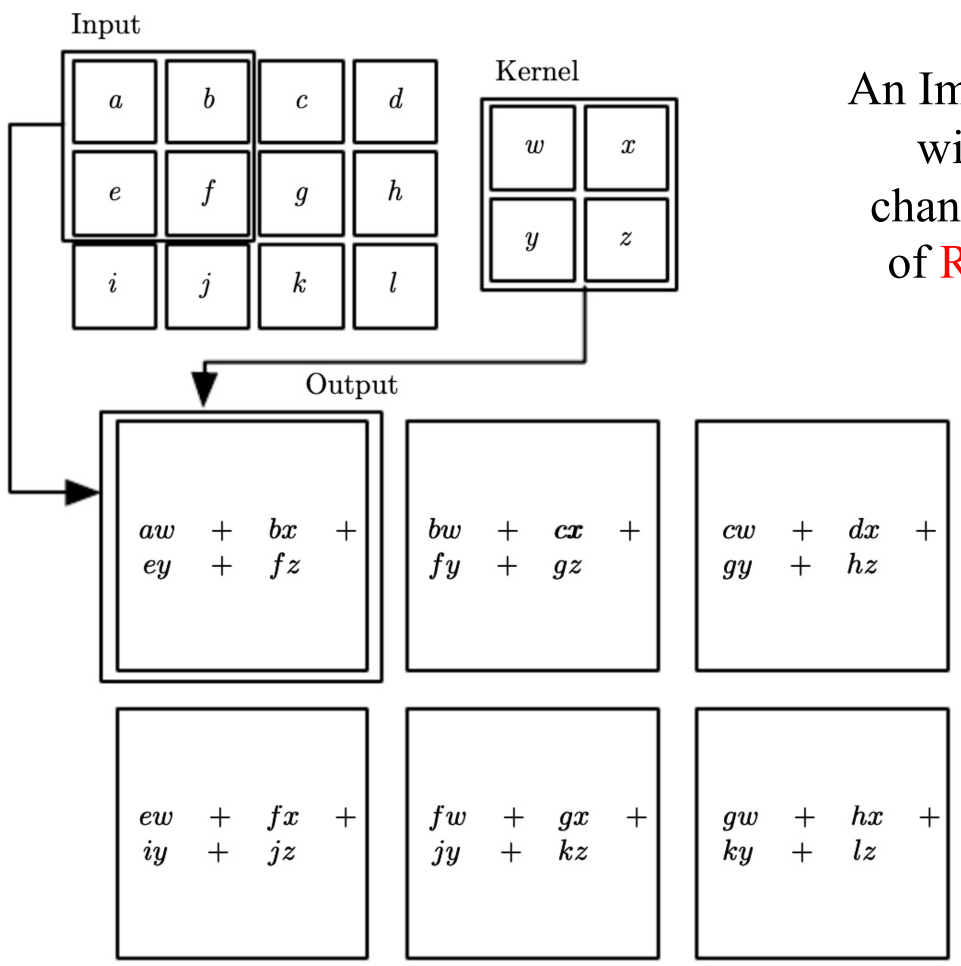
Output *Label*



AncoraSIR.com

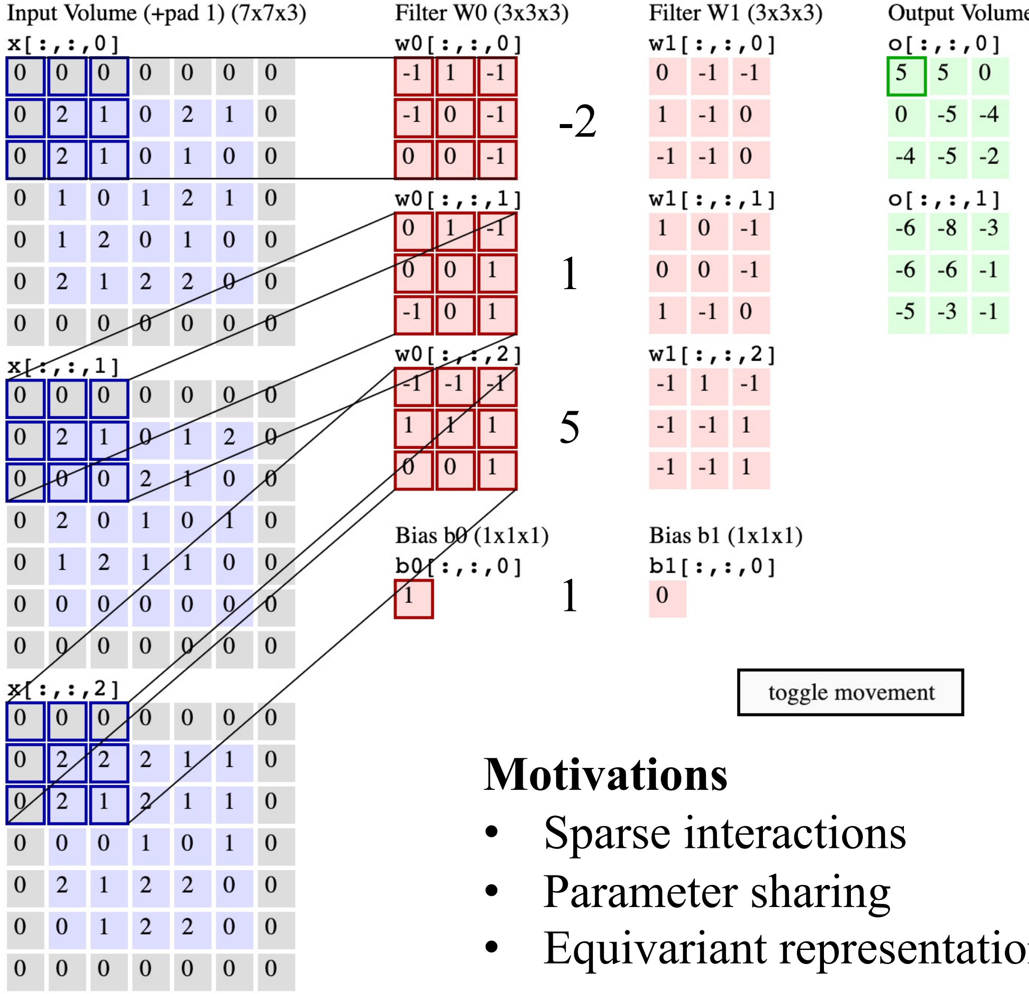
Convolutional Operation

$$s(t) = \int x(a)w(t-a)da = (x*w)(t)$$



AncoraSIR.com

An Image with 3 channels of RGB



Input Volume (+pad 1) (7x7x3)

x[:,:,0]

0	0	0	0	0	0	0
0	2	1	0	2	1	0
0	2	1	0	1	0	0
0	1	0	1	2	1	0
0	1	2	0	1	0	0
0	2	1	2	2	0	0
0	0	0	0	0	0	0

x[:,:,1]

0	0	0	0	0	0	0
0	2	1	0	1	2	0
0	0	0	2	1	0	0
0	2	0	1	0	1	0
0	1	2	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

x[:,:,2]

0	0	0	0	0	0	0
0	2	2	2	1	1	0
0	2	1	2	1	1	0
0	0	0	1	0	1	0
0	2	1	2	2	0	0
0	0	1	2	2	0	0
0	0	0	0	0	0	0

Filter W0 (3x3x3)

w0[:,:,0]

-1	1	-1
-1	0	-1
0	0	-1

w0[:,:,1]

0	1	-1
0	0	1
-1	0	1

w0[:,:,2]

-1	-1	-1
1	1	1
0	0	1

Bias b0 (1x1x1)

b0[:,:,0]

1

Filter W1 (3x3x3)

w1[:,:,0]

0	-1	-1
1	-1	0
-1	-1	0

w1[:,:,1]

1	0	-1
0	0	-1
1	-1	0

w1[:,:,2]

-1	1	-1
-1	-1	1
-1	-1	1

Bias b1 (1x1x1)

b1[:,:,0]

0

Output Volume (3x3x2)

o[:,:,0]

5	5	0
0	-5	-4
-4	-5	-2

o[:,:,1]

-6	-8	-3
-6	-6	-1
-5	-3	-1

toggle movement

-2

1

5

1

-2+1+5+1

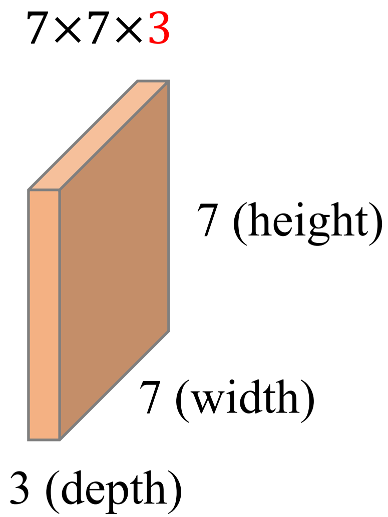
Motivations

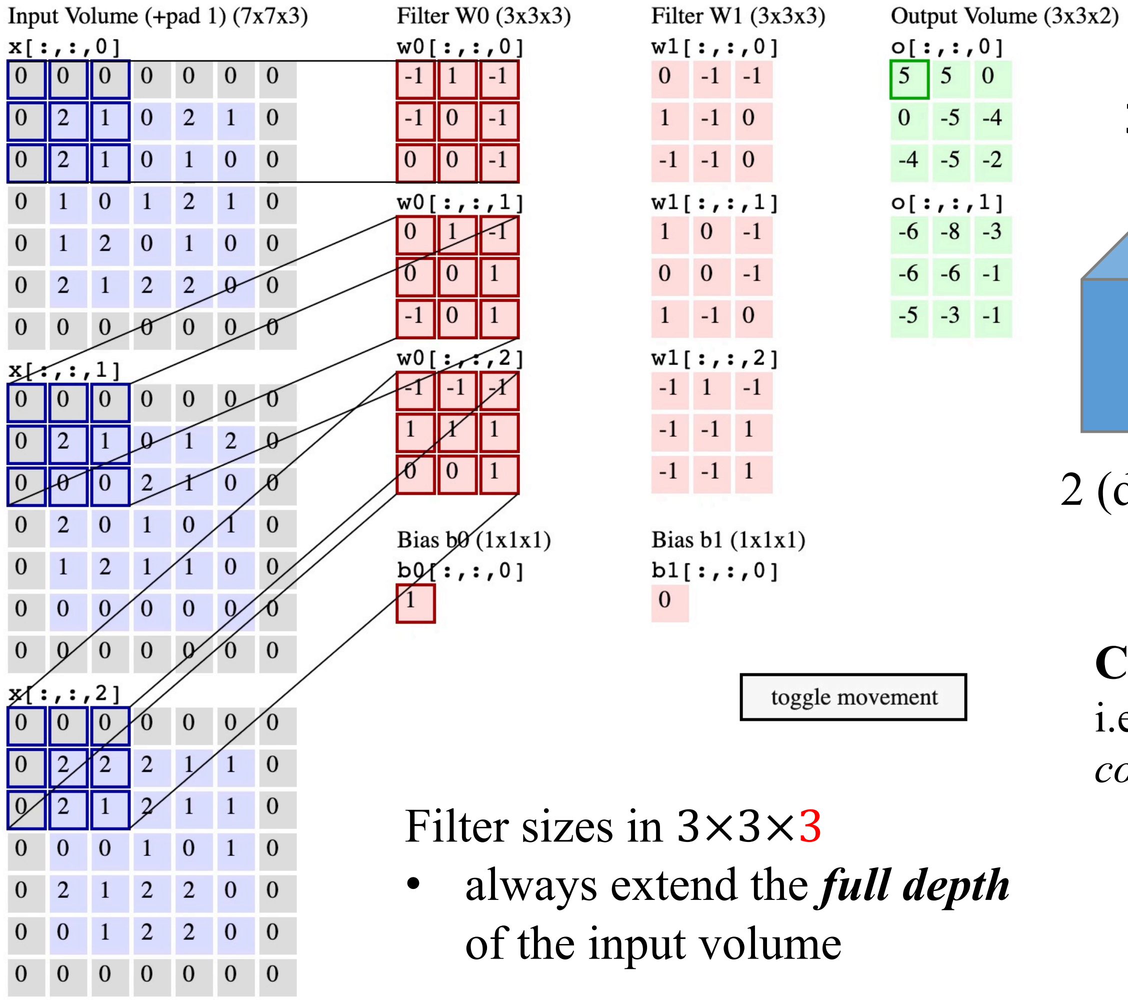
- Sparse interactions
- Parameter sharing
- Equivariant representations

Convolution in 3D Volumes

Preserved spatial structure between the input and output volumes in width, height, number of channels

Input Volume in 3D





Input Volume (+pad 1) (7x7x3)

x[:,:,0]

0	0	0	0	0	0	0
0	2	1	0	2	1	0
0	2	1	0	1	0	0
0	1	0	1	2	1	0
0	1	2	0	1	0	0
0	2	1	2	2	0	0
0	0	0	0	0	0	0

x[:,:,1]

0	0	0	0	0	0	0
0	2	1	0	1	2	0
0	0	0	2	1	0	0
0	2	0	1	0	1	0
0	1	2	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

x[:,:,2]

0	0	0	0	0	0	0
0	2	2	2	1	1	0
0	2	1	2	1	1	0
0	0	0	1	0	1	0
0	2	1	2	2	0	0
0	0	1	2	2	0	0
0	0	0	0	0	0	0

Filter W0 (3x3x3)

w0[:,:,0]

-1	1	-1
-1	0	-1
0	0	-1

w0[:,:,1]

0	1	-1
0	0	1
-1	0	1

w0[:,:,2]

-1	-1	-1
1	1	1
0	0	1

Bias b0 (1x1x1)

b0[:,:,0]

1

Filter W1 (3x3x3)

w1[:,:,0]

0	-1	-1
1	-1	0
-1	-1	0

w1[:,:,1]

1	0	-1
0	0	-1
1	-1	0

w1[:,:,2]

-1	1	-1
-1	-1	1
-1	-1	1

Bias b1 (1x1x1)

b1[:,:,0]

0

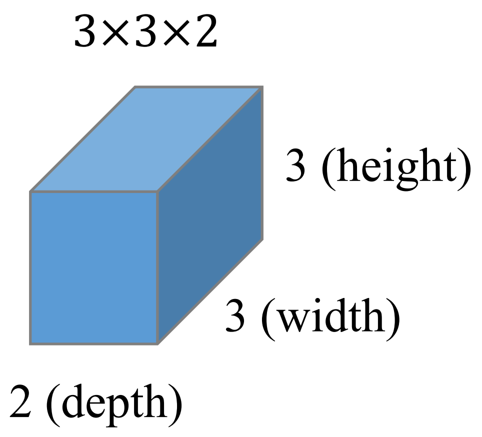
Output Volume (3x3x2)

o[:,:,0]

5	5	0
0	-5	-4
-4	-5	-2

o[:,:,1]

-6	-8	-3
-6	-6	-1
-5	-3	-1



Output Volume in 3D

Layers in a ConvNet: Transform an input 3D volume to an output 3D volume with some differentiable function that may or may not have parameters.

Filter sizes in 3×3×3

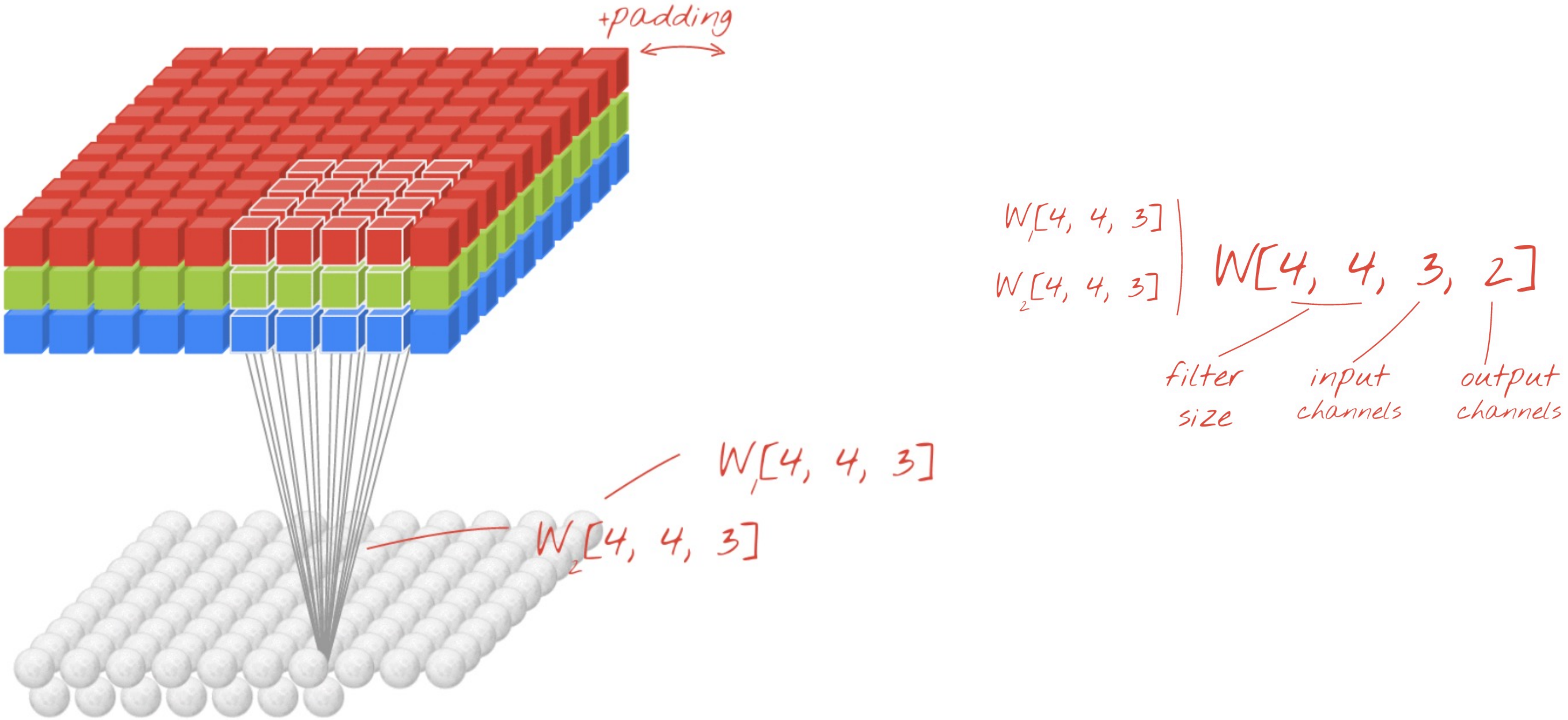
- always extend the ***full depth*** of the input volume

Convolve the filter with the image i.e. "*slide over the image spatially, computing dot products*"

AncoraSIR.com

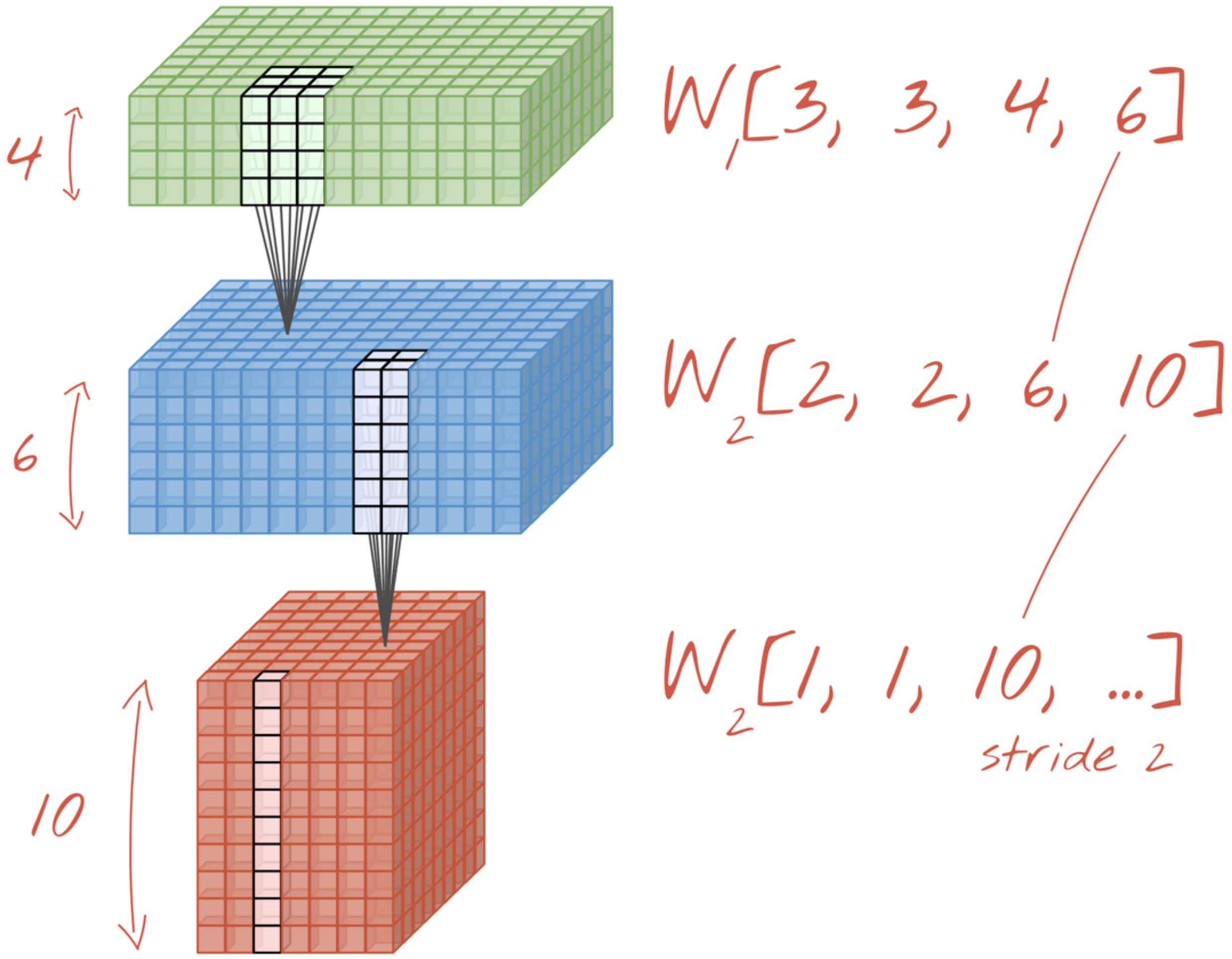
The Design of a Convolutional Layer

Defined by the filter (or kernel) size, the number of filters applied and the stride



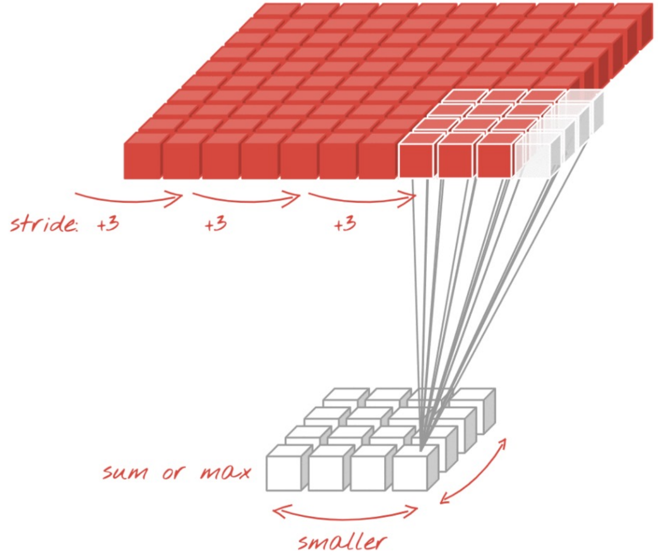
Output Volume Size

Defined by the filter (or kernel) size, the number of filters applied and the stride



AncoraSIR.com

- Depth (number of channels):
  - *adjusted by using more or fewer filters*
- Width & Height:
  - *adjusted by using a stride >1*
  - *(or with a max-pooling operation)*



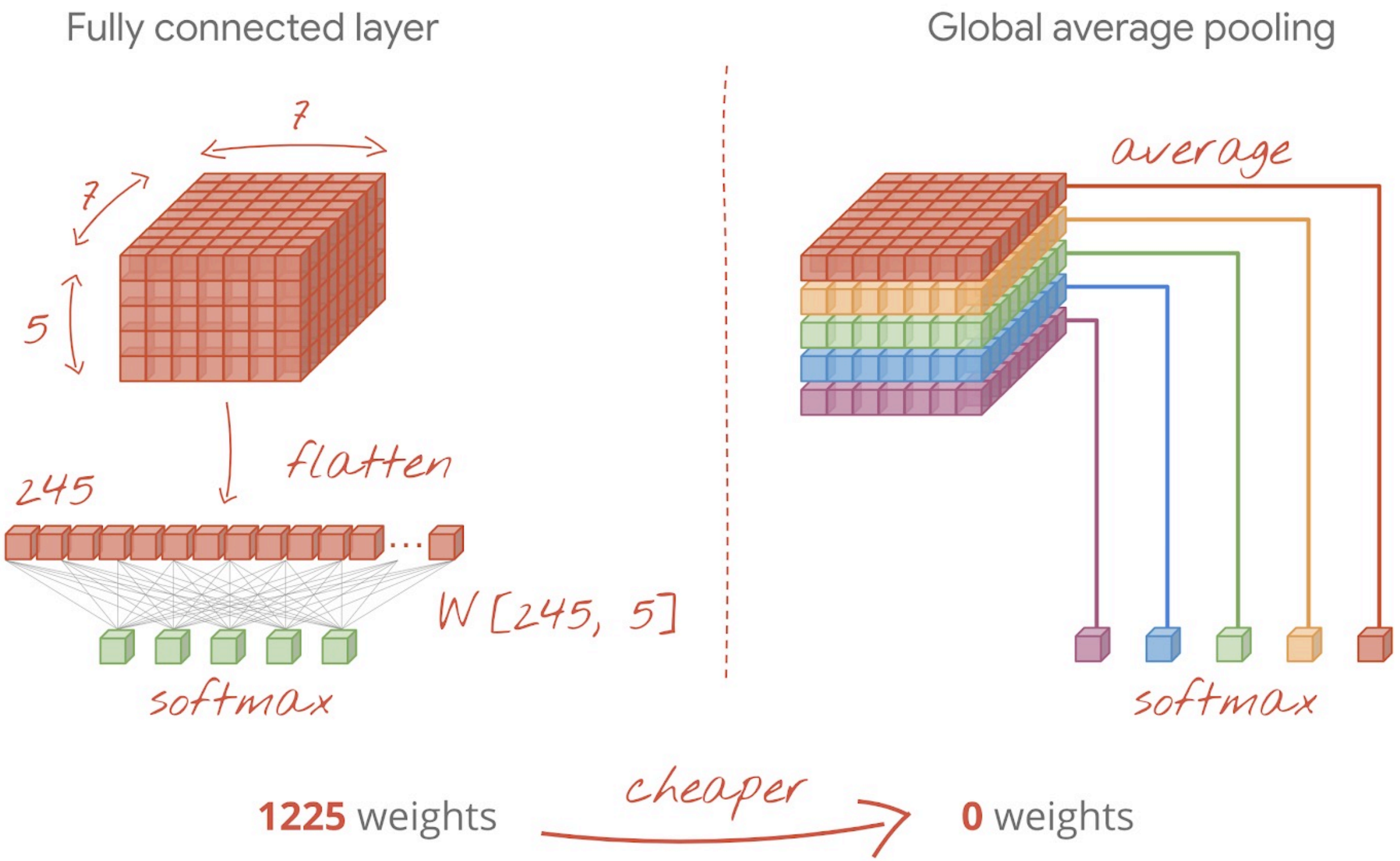
The Last Layer

From a Cubic Volume in 3D to predicted labels

Similar like a normal neural network

Expensive in #weights

But preseves the location data (x, y)



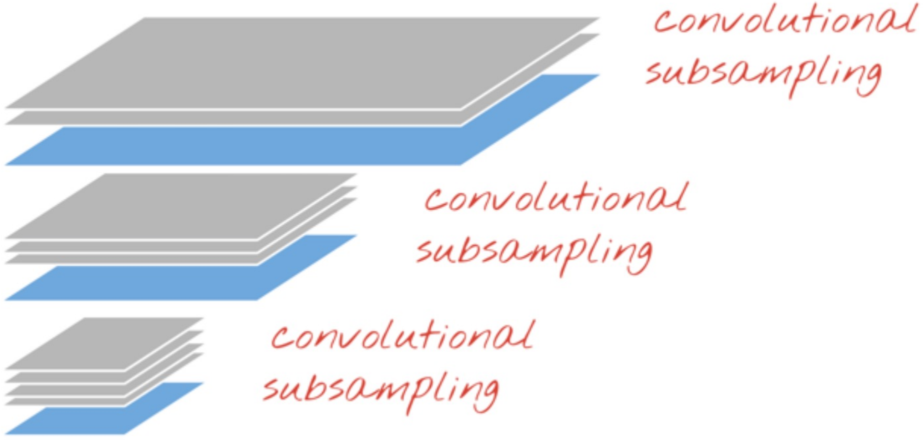
Much lighter in calculation

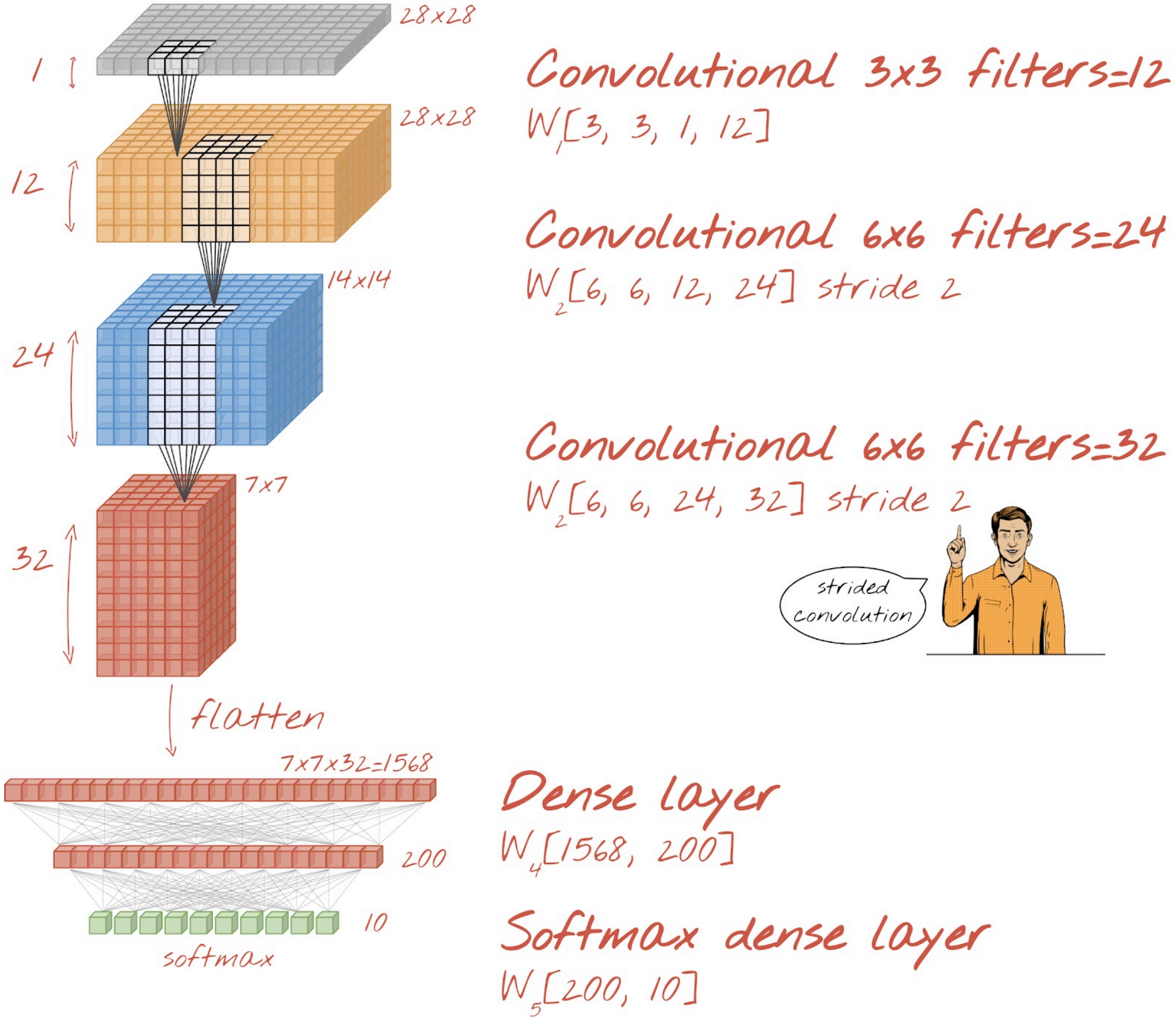
The average pooling explicitly discards all location data

AncoraSIR.com

Stacking Up a ConvNet

Layer-by-layer

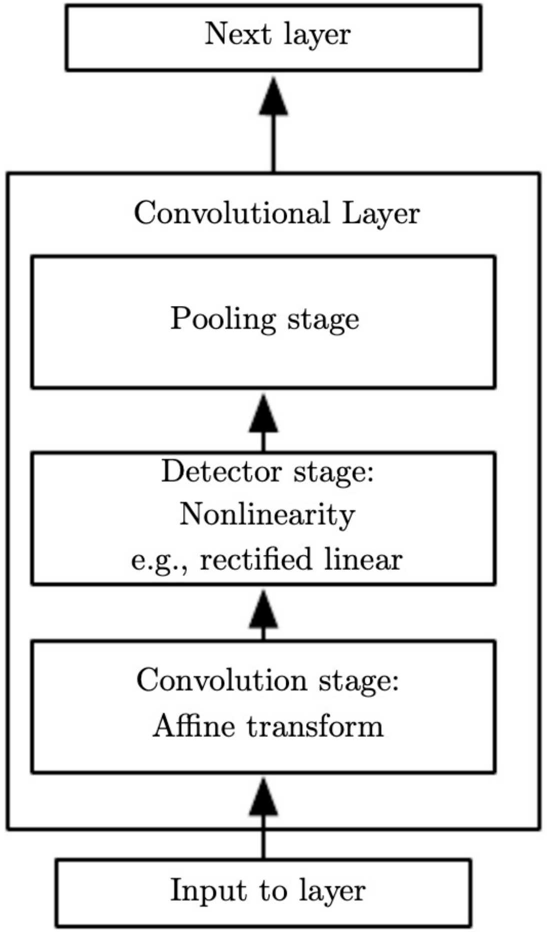




Layers in ConvNets

The Three Stages of a Typical ConvNet Layer

The Convolution, Detector and Pooling Stages



- The maximum output within a rectangular neighborhood (max-pooling)
- The average of a rectangular neighborhood
- The L2 norm of a rectangular neighborhood
- A weighted average based on the distance from the central pixel

Replace the output of the net at a certain location with a summary statistic of the nearby outputs (can be viewed as a further abstraction of the learned features)

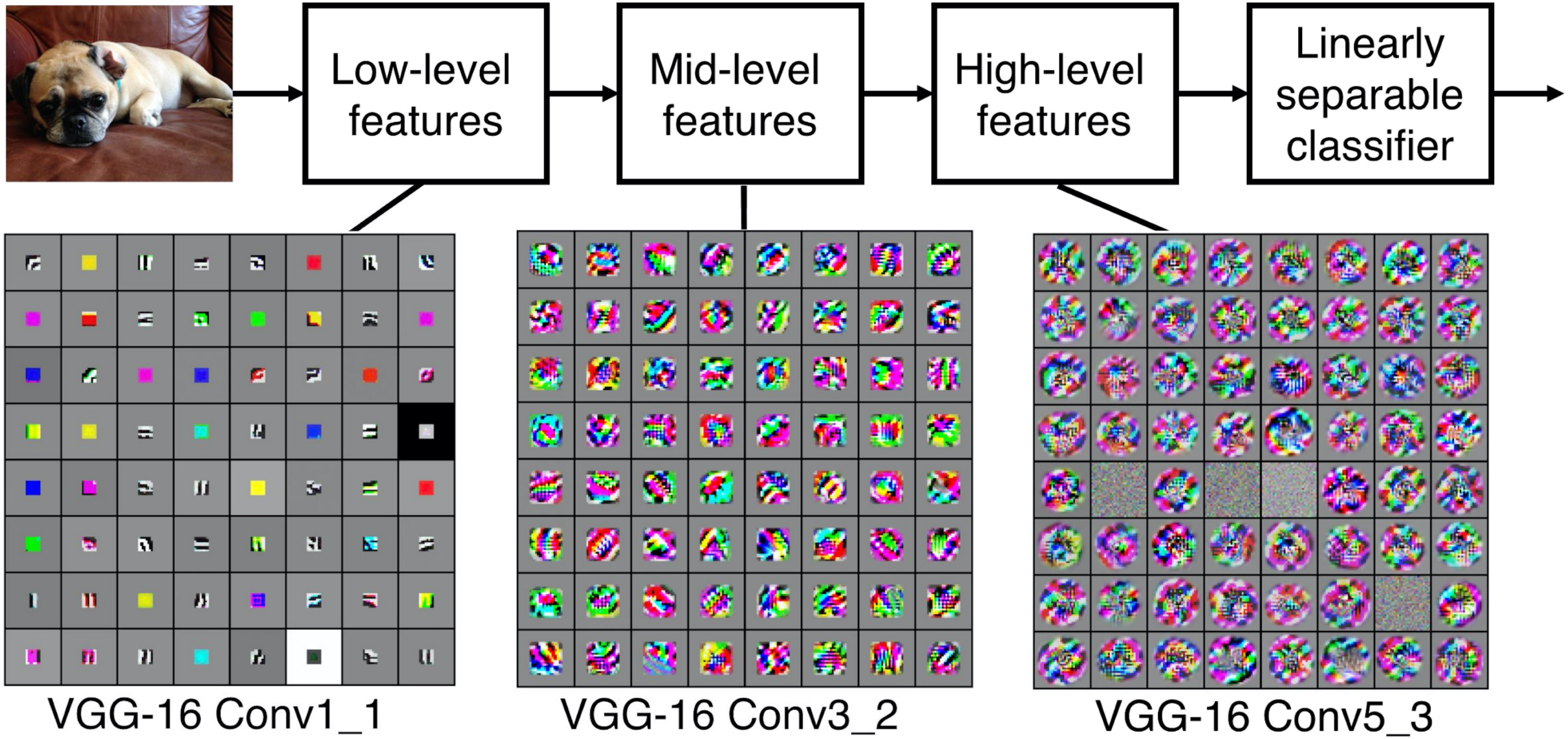
Each linear activation is run through a nonlinear activation function, such as ReLU (can be viewed as activation function)

Performs several convolutions in parallel to produce a set of linear activations (can be viewed as weighted-sum)

AncoraSIR.com

A Visualized Understanding of ConvNet

Multi-layered abstraction of 3D features towards a linerly separable classification



AncoraSIR.com

https://arxiv.org/pdf/1311.2901.pdf

A Simple ConvNet for CIFAR-10 Classification

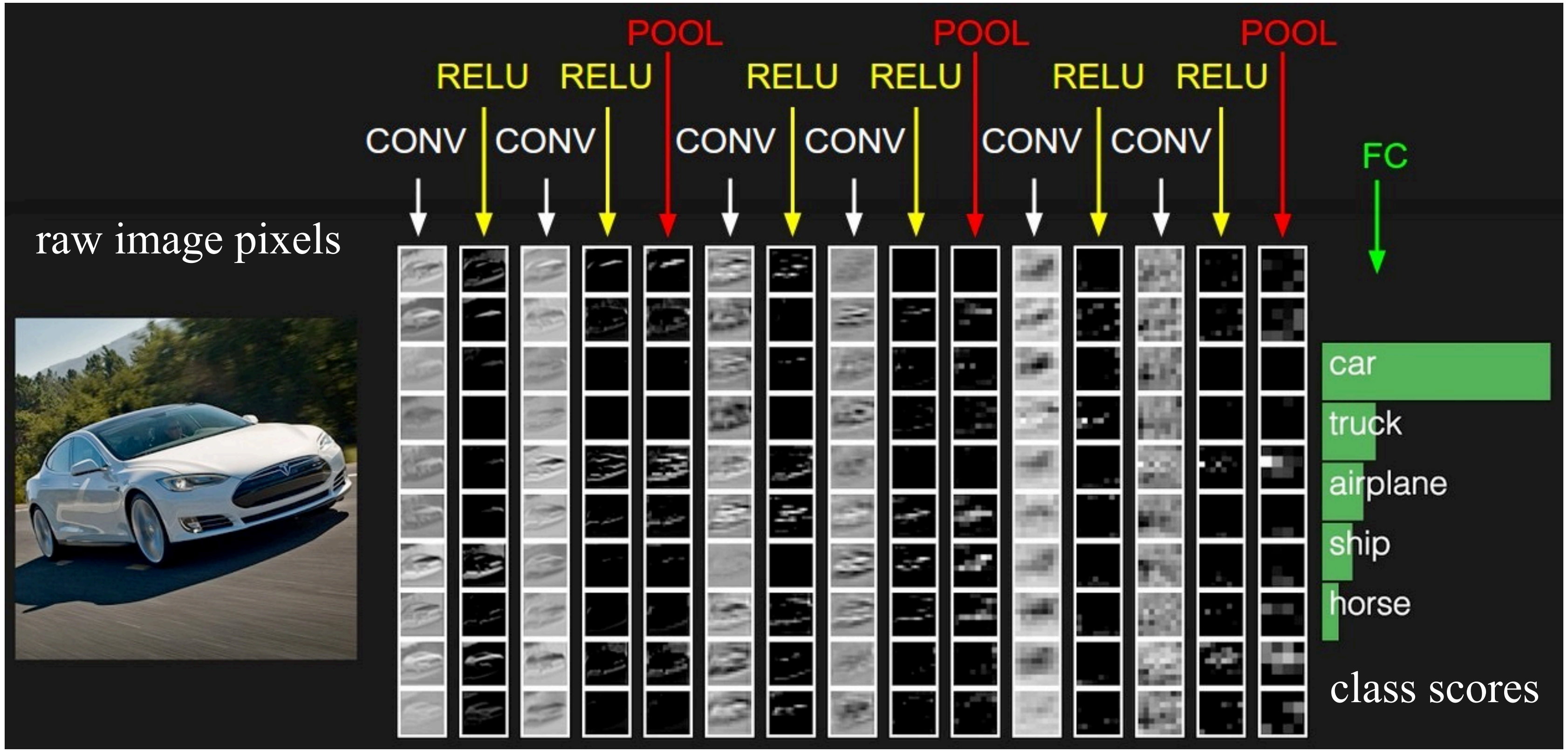
[INPUT - CONV - RELU - POOL - FC]

CONV layer compute the output of neurons that are connected to local regions in the input, i.e. [32x32x12] with 12 filters.

RELU layer will apply an elementwise activation function, such as the max(0,x) thresholding at zero. This leaves the size of the volume unchanged ([32x32x12]).

POOL layer will perform a downsampling operation along the spatial dimensions (width, height), resulting in volume such as [16x16x12].

INPUT layer [32x32x3] will hold the raw pixel values of the image



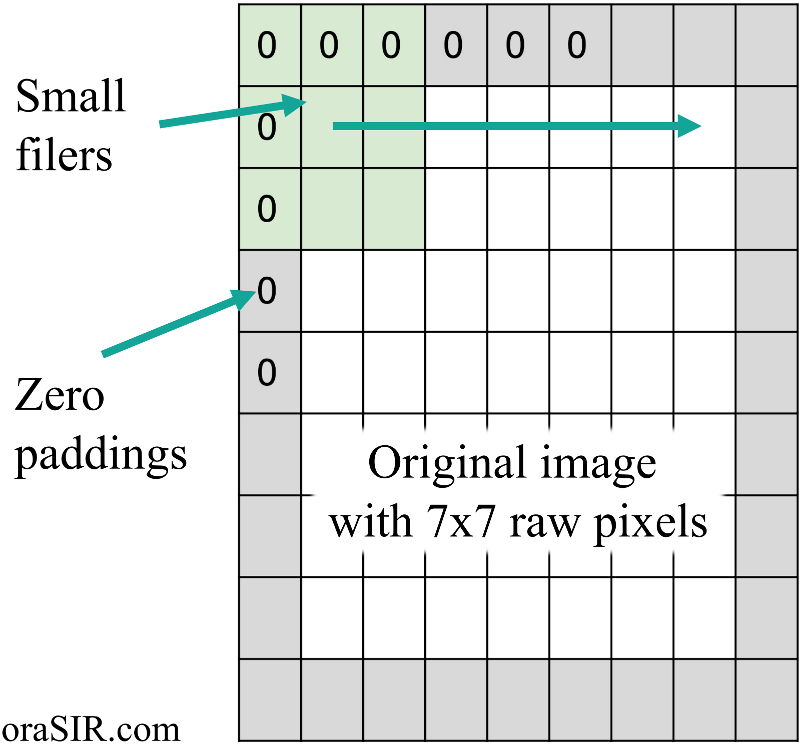
FC (i.e. fully-connected) layer will compute the class scores, resulting in volume of size [1x1x10], where each of the 10 numbers correspond to a class score

AncoraSIR.com

Convolutional Layer

Small filters that slide across the input volume

- Small-size filers
  - e.g. 3x3 or at most 5x5, using a stride of S=1,
  - Padding the input volume with zeros to avoid altering the spatial dimensions of the input.



INPUT features: 7x7

Filer size: 3x3

Stride: 1 (move step-by-step)

Padding: 1 pixel of 0 on all borders

OUTPUT features: 7x7

What if without paddings on the border?

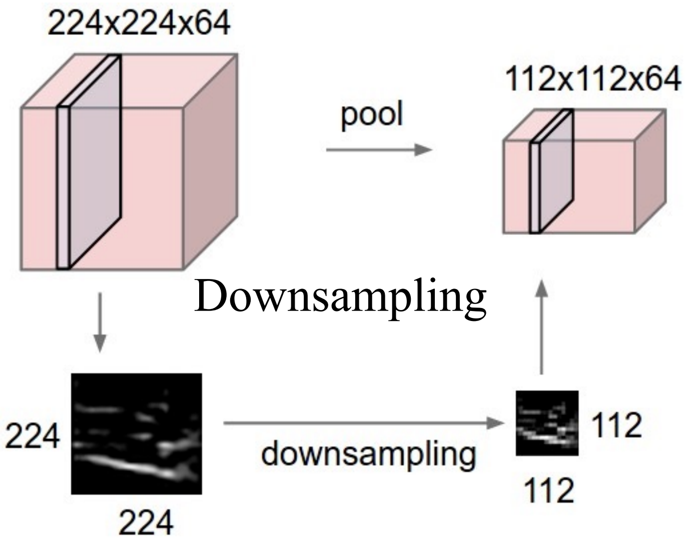
- *The spatial dimensions of the input will be changed, causing information loss on the border*

AncoraSIR.com

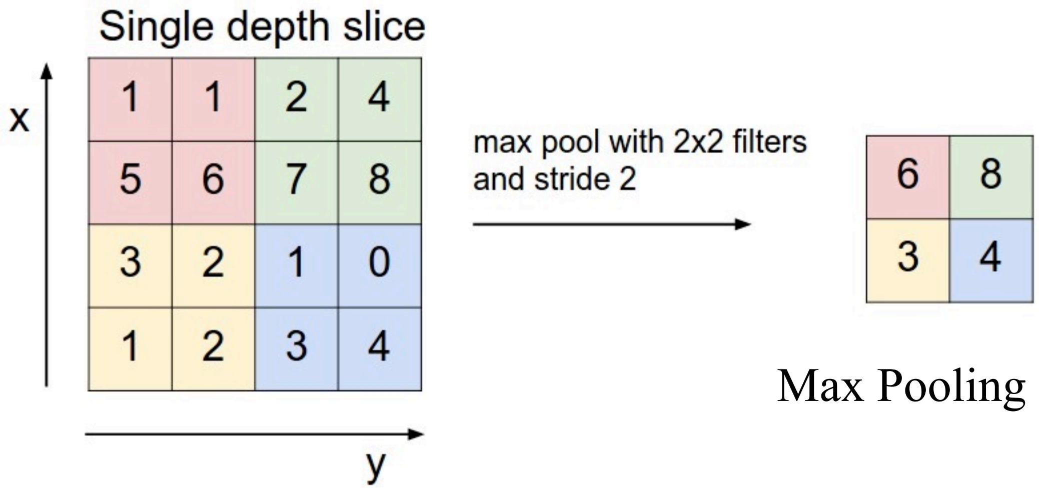
Pooling Layer

Downsampling the spatial dimensions of the input volume

- A network-wise regularization
  - Progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network, and hence to also control overfitting
  - Operates over each activation map independently
  - Usually, no need to zero padding (no convolutional operations)



Downsampling



1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

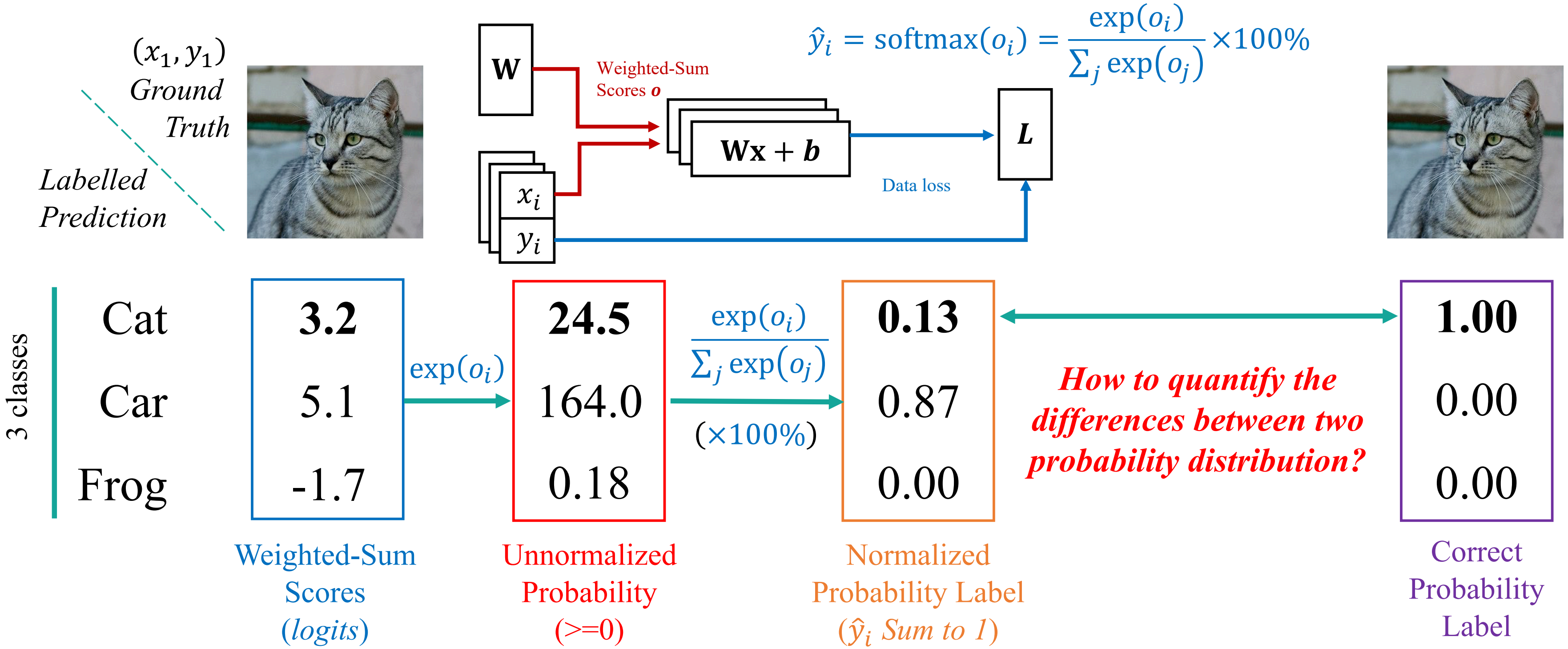
6	8
3	4

Max Pooling

Fully-Connected Layer

Full connections to all activations in the previous layer, as seen in regular Neural Networks

- Contains neurons that connect to the entire input volume
- Softmax is a common choice



$$\hat{y}_i = \text{softmax}(o_i) = \frac{\exp(o_i)}{\sum_j \exp(o_j)} \times 100\%$$

3 classes	Weighted-Sum Scores *(logits)*	Unnormalized Probability (>=0)	Normalized Probability Label *($\hat{y}_i$ Sum to 1)*	Correct Probability Label
Cat	**3.2**	**24.5**	**0.13**	**1.00**
Car	5.1	164.0	0.87	0.00
Frog	-1.7	0.18	0.00	0.00

How to quantify the differences between two probability distribution?

ConvNet Architectures

Common choice of hyperparameters of ConvNet designs

- $\text{INPUT} \rightarrow \left[[\text{CONV} \rightarrow \text{RELU}] * \text{N} \rightarrow \text{POOL?}\right] * \text{M} \rightarrow [\text{FC} - \text{RELU}] * \text{K} \rightarrow \text{FC}$
  - the * indicates repetition,
  - the POOL? indicates an optional pooling layer.
  - N >= 0 (and usually N <= 3), M >= 0, K >= 0 (and usually K < 3)

- **INPUT** (that contains the image) should be divisible by 2 many times
  - 32 (e.g. CIFAR-10), 64, 96 (e.g. STL-10), or 224 (e.g. ImageNet), 384, and 512
- **CONV** should be using small filters using a stride of S=1
  - 3x3 or at most 5x5 with zero padding of the input volume
- **POOL** downsamples the spatial dimensions of the input
  - Common setting is to use max-pooling with 2x2 receptive fields with a stride of 2

AncoraSIR.com

Thank you~

songcy@sustech.edu.cn